CLAIMS

What is claimed is:

1. A method for controlling the value of a RAM variable inside an executable program, comprising:

presenting a software program in executable form and having a plurality of machine instructions of a finite quantity of fixed lengths;

identifying at least one machine instruction that accesses a variable defined in random access memory associated with the software program;

defining a replacement instruction for the at least one machine instruction; and

replacing the at least one machine instruction in the executable form of the software program with the replacement instruction.

2. The method of Claim 1 wherein the replacement instruction is further defined as a branch instruction that references an address outside an address space for the software program or

3. The method of Claim 1 wherein the replacement instruction is further defined as a no operation instruction.

4. The method of Claim 1 wherein the replacement instruction is further defined as a instruction that can cause at least one of an interrupt and an exception to occur in the microprocessor.

5. The method of Claim 1 wherein the step of identifying at least one machine instruction further comprises determining location information for the at least one machine instruction within the software program.

6. The method of Claim 5 wherein the step of determining location information further comprises identifying an address for the at least one machine instruction using the image of the executable containing the machine instructions that comprise the executable.

7. The method of Claim 6 wherein the step of replacing the at least one machine instruction further comprises inserting the replacement instruction into a program memory image of the software program at said address.

8. The method of Claim 2 wherein said branch instruction references a set of relocation instruction residing outside an address space for the software program.

9. The method of Claim 1 further comprises executing the executable form of the software program having the replacement instruction.

10. A computer-implemented calibration system for modifying RAM variables of a software program embedded in a microprocessor, comprising:

an instruction locator adapted to receive an address for RAM variable within an software program and operable to identify at least one machine instruction in an executable form of the software program that accesses the RAM variable; and

an instruction replacement component in data communication with the instruction locator, the instruction replacement component adapted to receive a replacement instruction for the at least one machine instruction and operable to replace the at least one machine instruction in the executable form of the software program with the replacement instruction.

11. The computer-implemented system of Claim 10 wherein the instruction locator is operable to identify an address for the specified machine instruction using the image of the executable containing the machine instructions that comprise the executable.

12. The computer-implemented system of Claim 11 wherein the instruction replacement component is operable to insert the replacement instruction into a program memory image of the software program at said address.

13. The computer-implemented system of Claim 10 wherein the replacement instruction is further defined as a branch instruction that references an address outside an address space for the software program.

14. The computer-implemented system of Claim 10 wherein the instruction replacement component is operable to generate a set of relocation instructions, such that the replacement instruction passes processing control to the set of relocation instructions.

15. The computer-implemented system of Claim 14 wherein the instruction replacement component is further operable to insert the set of relocation instructions in a memory space of the microprocessor that resides outside of an address space for the software program.